PathRelativePathTo

The destination string buffer must be long enough to hold the return file path

Sean Barnum, Cigital, Inc. [vita[1]]

2007-04-02

Part "Original Cigital Coding Rule in XML"

Mime-type: text/xml, size: 4617 bytes

<table>
<tr><td><b>Attack Category</b></td><td>• Malicious Input</td></tr>
<tr><td><b>Vulnerability Category</b></td><td>• Buffer Overflow
• Unconditional</td></tr>
<tr><td><b>Software Context</b></td><td>• File Path Management</td></tr>
<tr><td><b>Location</b></td><td>• shlwapi.h</td></tr>
<tr><td><b>Description</b></td><td>The destination string buffer for the PathRelativePathTo() function must be long enough to hold the return file path.
The function PathRelativePathTo() function takes a pair of paths and generates a relative path from one to the other. This could, in theory, be the entirety of one input string, plus "\.." units for each directory in the other input string.
It is undefined what will occur if both the paths are MAX_PATH in length and have nothing in common, or even worse if the "from" path contains many single character directory names causing each of them to expand to two characters (".."). This leaves the potential for returned paths of an unknown state and content.</td></tr>
<tr><td><b>APIs</b></td><td><table>
<tr><th>Function Name</th><th>Comments</th></tr>
<tr><td>PathRelativePathTo</td><td></td></tr>
<tr><td>PathRelativePathToA</td><td>ANSII implementation</td></tr>
<tr><td>PathRelativePathToW</td><td>Unicode implementation</td></tr>
</table></td></tr>
<tr><td><b>Method of Attack</b></td><td>If the two paths are long and have little in common, the resulting path could be quite large. If the attacker purposely provides long path names, he could overrun a buffer that is not at least MAX_PATH in size.</td></tr>
<tr><td><b>Exception Criteria</b></td><td></td></tr>
<tr><td><b>Solutions</b></td><td><table>
<tr><th>Solution Applicability</th><th>Solution Description</th><th>Solution Efficacy</th></tr>
</table></td></tr>
</table>

[1] http://buildsecurityin.us-cert.gov/bsi/about_us/authors/35-BSI.html (Barnum, Sean)

<table>
<tr><td></td><td><table><tr><td>Whenever PathRelativePathTo() is called.</td><td>The first parameter, pszPath, must be at least MAX_PATH characters in length.</td><td>Effective.</td></tr></table></td></tr>
<tr><td><b>Signature Details</b></td><td>BOOL PathRelativePathTo(
LPTSTR pszPath,
LPCTSTR pszFrom,
DWORD dwAttrFrom,
LPCTSTR pszTo,
DWORD dwAttrTo
);</td></tr>
<tr><td><b>Examples of Incorrect Code</b></td><td><code>TCHAR szOut[10] = TEXT(""); // Buffer is too small
TCHAR szFrom[ ] = TEXT("c:\\a\\b\\path");
TCHAR szTo[ ] = TEXT("c:\\a\\x\\y\\file");

if (!PathRelativePathTo(szOut, szFrom, FILE_ATTRIBUTE_DIRECTORY, szTo, FILE_ATTRIBUTE_NORMAL))
{ handleError(); }</code></td></tr>
<tr><td><b>Examples of Corrected Code</b></td><td><code>TCHAR szOut[MAX_PATH] = ""; // Buffer is correctly sized
TCHAR szFrom[ ] = TEXT("c:\\a\\b\\path");
TCHAR szTo[ ] = TEXT("c:\\a\\x\\y\\file");

if (!PathRelativePathTo(szOut, szFrom, FILE_ATTRIBUTE_DIRECTORY, szTo, FILE_ATTRIBUTE_NORMAL))
{ handleError(); }</code></td></tr>
<tr><td><b>Source Reference</b></td><td>• http://msdn.microsoft.com/library/default.asp?url=/library/en-us/shellcc/platform/shell/reference/shlwapi/path/pathrelativepathto.asp[2]</td></tr>
<tr><td><b>Recommended Resource</b></td><td></td></tr>
<tr><td><b>Discriminant Set</b></td><td><table><tr><td><b>Operating System</b></td><td>• Windows</td></tr><tr><td><b>Languages</b></td><td>• C
• C++</td></tr></table></td></tr>
</table>

Cigital, Inc. Copyright

The Build Security In (BSI) portal is sponsored by the U.S. Department of Homeland Security (DHS), National Cyber Security Division. The Software Engineering Institute (SEI) develops and operates BSI. DHS funding supports the publishing of all site content.

[1] mailto:copyright@cigital.com